**LESSON:**

HTML – stands for **H**yper**T**ext **M**arkup **L**anguage

XHTML stands for E**X**tensible **H**yper**T**ext **M**arkup **L**anguage

* XHTML is almost identical to HTML
* XHTML is stricter than HTML
* XHTML is HTML defined as an XML application
* XHTML is supported by all major browsers

**Why XHTML?**

Many pages on the internet contain "bad" HTML.

This HTML code works fine in most browsers (even if it does not follow the HTML rules):

For Example:

<html>  
<head>  
  <title>This is bad HTML</title>  
  
<body>  
  <h1>Bad HTML  
  <p>This is a paragraph  
</body>

The code above is missing two close tags </head> & </html>. It will run on most browsers even if it contains errors. XHTML resolves that problem. **Don’t worry about knowing the above code, it will be discussed later.**

Today's market consists of different browser technologies. Some browsers run on computers, and some browsers run on mobile phones or other small devices. Smaller devices often lack the resources or power to interpret "bad" markup (tags). That is why the switch to XHTML.

XML is a markup language where documents must be marked up correctly (be "well-formed").

By combining the strengths of HTML and XML, XHTML was developed.

XHTML is HTML redesigned as XML.

**HTML 4 & XHTML** use precisely the same elements, attributes and values. The difference is the syntax.

**Note:** (X)HTML means it is the same for both HTML and XHTML. When talking about XHTML, I’ll use XHTML without the parenthesis. I’ll use HTML when just talking about HTML only.

**(X)HTML** consists of tags. Tags can be identified by open & closed angle brackets (<…>) open. Some tags have a closed tag and can be identified with a slash (</ … >) closed tag. While other tags (empty tags) don’t need closed tags because it doesn’t wrap around a text but does consist of a slash at the end of tag command (<…/>) empty tag.

XHTML is stricter than HTML. Browsers are more sensitive with proper structure for XHTML. All attributes for XHTML must be enclosed in double quotes. In HTML, forgetting to enclose attributes in double quotes does not cause an error. XHTML is case sensitive and all elements, values & attributes must be lowercase. HTML is not case sensitive.

So why XHTML! It will be easier to format with CSS or generate from or convert into a database, and adapt to other systems.

**Examples:**

<em>Text</em> - This will italicize all text between the **open** <em> **tag** and the **closed** </em> **tag**

**Empty elements** – meaning they stand alone unlike the above example don’t need a closed tag.

<img src = “picture.jpg” width = “300” /> - Displays a picture has no effect on any text

Some tags have **attributes** that effects how the tag will behaves. “src” & “width” in the above img tag are the attributes and has a **value** of 300. The **command** using the above example is img & em

Right now don’t worry about knowing the tags. We are just covering the structure of a tag. You will learn what the tags are in future lessons and Labs.

**Let’s review:**

**Tags** are in open and closed angle brackets <> also called **open tag** it starts the command at that point.

**Closed Tags** can be identified with the slash inside of the double <>’s. (</… >). They stop the command at that point.

**Empty elements** – are stand-alone tags that don’t need a closed tag.

**Command** follows the < angle bracket. It applies that command at that point until it is closed.

**Attributes** – are in the open tag after the command and has an effect on how the command will be displayed or act.

**Value** – follows the attribute with an equal (=) symbol.

**Getting started**

First you will need a word processor. Notepad, WordPad, etc…

**Saving a Web Page**

When saving your web page, you must save as **.htm** or **.html** extensions.

**Four Major Tags**

There are four major tags of a web page.

1. **<html> ….. </html> -** opens an HTML document for Browsers to interpret. All tags and text are located between these two tags
2. **<head> …. </head> -** Opens up the heading so a title can be put in your page. Located between the <html>’s tags
3. **<title> - This is where you put the title of the page and it will show above the toolbar not on the main screen area. </title>.** The <title> & </title> is located between the open and closed tags of the <head> tags.
4. **<body> This is where most of your text and code goes</body>.** The <body> tags are located between the <html> & </html> tags.

**The Most Important Differences from HTML:**

**Document Structure**

* XHTML DOCTYPE is **mandatory**
* The xmlns attribute in <html> is **mandatory**
* <html>, <head>, <title>, and <body> are **mandatory**

**XHTML Elements**

* XHTML elements must be **properly nested**
* XHTML elements must always be **closed**
* XHTML elements must be in **lowercase**
* XHTML documents must have **one root element**

**XHTML Attributes**

* Attribute names must be in **lower case**
* Attribute values must be **quoted**
* Attribute minimization is **forbidden**

**Again, don’t worry about the above statements; we will be covering all of this throughout this course.**

**Quick Lesson on Docs**

**<!DOCTYPE ....> Is Mandatory**

The <!DOCTYPE> declaration must be the very first thing in your HTML document, before the <html> tag.

The <!DOCTYPE> declaration is not an HTML tag; it is an instruction to the web browser about what version of HTML the page is written in.

An XHTML document must have an XHTML DOCTYPE declaration. The <!DOCTYPE> declaration is NOT case sensitive.

The <html>, <head>, <title>, and <body> elements must also be present, and the xmlns attribute in <html> must specify the xml namespace for the document.

This example shows an XHTML document with a minimum of required tags:

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

There are several to pick from

* 1. XHTML 1.0 Strict

This DTD contains all HTML elements and attributes, but does NOT INCLUDE presentational or deprecated elements (like font). Framesets are not allowed. The markup must also be written as well-formed XML.

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

* 1. XHTML 1.0 Transitional

This DTD contains all HTML elements and attributes, INCLUDING presentational and deprecated elements (like font). Framesets are not allowed. The markup must also be written as well-formed XML.

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

* 1. XHTML 1.0 Frameset

This DTD is equal to XHTML 1.0 Transitional, but allows the use of frameset content.

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">

* 1. XHTML 1.1

This DTD is equal to XHTML 1.0 Strict, but allows you to add modules (for example to provide ruby support for East-Asian languages).

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

**Meta Tags Definition and Usage**

Metadata is data (information) about data.

The <meta> tag provides metadata about the HTML document. Metadata will not be displayed on the page, but will be machine parsable.

Meta elements are typically used to specify page description, keywords, author of the document, last modified, and other metadata.

The metadata can be used by browsers (how to display content or reload page), search engines (keywords), or other web services.

**Tips and Notes**

**Note:** <meta> tags always goes inside the <head> element.

**Note:** Metadata is always passed as name/value pairs.

**Note:** The content attribute MUST be defined if the name or the http-equiv attribute is defined. If none of these are defined, the content attribute CANNOT be defined.

**Examples**

**Example 1 - Define keywords for search engines:**

<meta name="keywords" content="HTML, CSS, XML, XHTML, JavaScript">

**Example 2 - Define a description of your web page:**

<meta name="description" content="Free Web tutorials on HTML and CSS">

**Example 3 - Define the author of a page:**

<meta name="author" content="Hege Refsnes">

**Example 4 - Refresh document every 30 seconds:**

<meta http-equiv="refresh" content="30">

**Comments: <!- - … - ->**

**Definition and Usage**

The comment tag is used to insert comments in the source code. Comments are not displayed in the browsers.

You can use comments to explain your code, which can help you when you edit the source code at a later date. This is especially useful if you have a lot of code.